

Комп'ютерний практикум № 2. Шаблони проектування. Структурні шаблони

Мета:

- ознайомитися з основними шаблонами проектування, навчитися застосовувати їх при проектуванні і розробці ПЗ.

Теоретичні основи

Шаблон проектування або патерн (англ. Design pattern) в розробці програмного забезпечення - повторювана архітектурна конструкція, що представляє собою рішення проблеми проектування в рамках деякого часто виникаючого контексту.

Патерн - це не закінчене архітектурне рішення, яке можна безпосередньо перетворити в вихідний або машинний код. Це опис підходу до вирішення проблеми, який можна застосовувати в різних ситуаціях.

Це опис взаємодії об'єктів, класів, адаптованих для вирішення загальної задачі проектування в конкретному контексті.

Патерни - це прості приклади, які показують правильні способи організації взаємодії між класами і об'єктами.

Патерни описують:

- Правильні способи формування внутрішнього стану (полів) і поведінки (методів) об'єкту або класу
- Правильні способи створення об'єкта (через виклик конструктора або іншим способом)
- Правильні способи об'єднання об'єктів всередині групи
- Правильні способи організації інформаційних потоків (виклик методів і черговість викликів), що дозволяють налагодити гармонійну взаємодію між об'єктами і групами цих об'єктів в об'єктно-орієнтованих системах

Зазвичай шаблон не є закінченим зразком, який може бути прямо перетворений в код. Це приклади розв'язання задач, які можна використовувати в різних ситуаціях. Об'єктно-орієнтовані шаблони показують відносини і взаємодію між класами або об'єктами, без визначення того, які саме кінцеві класи або об'єкти додатки будуть використовуватися.

Будь-який шаблон проектування може стати палицею з двома кінцями: якщо він буде застосований не до місця, це може обернутися катастрофою і створити багато проблем в подальшому. У той же час, реалізований в потрібному місці, в потрібний час, він може стати справжнім рятівником.

Існує три основних види шаблонів проектування:

- **Породжуючі патерни.** Це патерни, які абстрагують процес породження класів і об'єктів.
 - Абстрактна фабрика (Abstract Factory)
 - Будівник (Builder)
 - Фабричний метод (Factory Method)
 - Прототип (Prototype)
 - Одинак (Singleton)
- **Структурні патерни.** Розглядають, як класи і об'єкти утворюють більші структури - більш складні за характером класи і об'єкти.
 - Адаптер (Adapter)
 - Міст (Bridge)
 - Компоновщик (Composite)
 - Декоратор (Decorator)
 - Фасад (Facade)
 - Пристосуванець (Flyweight)
 - Заступник (Proxy)
- **Поведінкові патерни** - вони визначають алгоритми і взаємодію між класами і об'єктами, тобто їх поведінку.
 - Ланцюжок обов'язків (Chain of responsibility)
 - Команда (Command)
 - Інтерпретатор (Interpreter)
 - Ітератор (Iterator)
 - Посередник (Mediator)
 - Зберігач (Memento)
 - Спостерігач (Observer)
 - Стан (State)
 - Стратегія (Strategy)
 - Шаблонний метод (Template method)
 - Відвідувач (Visitor)

Постановка задачі комп'ютерного практикуму № 2

При виконанні комп'ютерного практикуму необхідно виконати наступні дії:

- 1) Вивчити структурні патерни. Знати загальну характеристику та призначення кожного з них, особливості реалізації кожного зі структурних патернів та випадки їх застосування.
- 2) Реалізувати задачу згідно варіанту, запропонованого нижче. Розробити інтерфейси та класи з застосування одного або декількох патернів. Повністю реалізувати методи, пов'язані з реалізацією обраного патерну.
- 3) Повністю описати архітектуру проекту (призначення методів та класів), особливості реалізації обраного патерну. Для кожного патерну необхідно вказати основні класи та їх призначення,
- 4) Навести UML-діаграму класів

Варіанти індивідуальних завдань:

- 1) Розробити структуру організації армії в грі фентезі. Армія може складатися з загонів ельфів, орків, мінотаврів, кентаврів, циклопів, драконів, гідр, лицарів. Армія може містити як загони, так і одиночних воїнів, загін може складатися з інших загонів і одиночних воїнів.
- 2) Створити набір класів, котрі реалізують наступні перетворення текстових даних с наступним збереженням в файл:
 - a. Переклад по словам з використанням Google Translate API (робочий приклад можна знати, наприклад, тут <https://stackoverflow.com/questions/2246017/using-google-translate-in-c-sharp>)
 - b. Шифрування даних будь-яким одним алгоритмом.В програмному застосунку повинні бути реалізовані функції запису текстових даних в файл після використання будь-якої комбінації перетворень в будь-якій послідовності та читання даних з файлу зі зворотнім перетворенням.
- 3) Реалізувати задачу роботи з касовими апаратами, де апарати можуть бути різних виробників. Кожен касовий апарат має унікальний номер, колекцію товарів та можливість додати товар, зберегти та роздрукувати чек.
- 4) Реалізувати задачу криптографічного захисту даних з використанням бібліотеки System.Security.Cryptography(приклад використання можна подивитись тут <https://docs.microsoft.com/ru-ru/dotnet/api/system.security.cryptography?view=net-5.0>). Система криптографічного захисту повинна реалізовувати шифрування даних с використанням обраної комбінації алгоритмів хешування, симетричного та асиметричного шифрування. Необхідно реалізувати й дешифрування (ключі необхідно запитувати у користувача).
- 5) Розробити структуру даних для зберігання інформації про країни, їх адміністративні частини і міста. Пошук необхідних даних.

- 6) Реалізувати задачу «Інтернет-магазин по продажу комп'ютерної техніки». Товари - комплектуючі. У кожного товару є категорія, назва, артикул та ціна. Для товару в категорії «Процесор» необхідно вказувати тактову частоту, кількість ядер тощо. І так само для товарів у інших категоріях вказуються свої параметри. Крім продажу окремих комплектуючих, інтернет-магазин може продавати системні блоки, котрі збираються з наявних комплектуючих, з надбавкою 15% за збірку або комп'ютери повністю (системний блок+монітор) з надбавкою 5%, або інші варіації з різною надбавкою.
- 7) Існує модель системи «Магазин квітів». Реалізувати задачу, що дозволяє додавати до букету новий атрибут (наприклад, стрічку з фірмовою назвою), використовуючи при цьому існуючу модель породження букетів.
- 8) Реалізувати задачу «Розрахунок щоденної норми споживання кілокалорій». Необхідно врахувати, що норма споживання кілокалорій залежить від типу статури, зросту, ваги, віку, статі, групи фізичної активності (низька, середня і висока активність).
- 9) Реалізувати задачу "Універсальна електронна карта". Необхідно реалізувати універсальну електронну карту, в якій є функції паспорта, страхового полісу, банківської карти та інше.
- 10) Необхідно реалізувати задачу формування штатного розкладу підприємства/підрозділу. Зберігається ієрархічна структура структурних підрозділів підприємства. Для кожного структурного підрозділу заданий код, найменування та перелік посад за штатним розкладом. Для кожної посади зберігається: найменування, кількість ставок у підрозділі, оклад. Повинні бути реалізовані можливості виведення штатного розкладу для будь-якого підрозділу, включаючи підлеглі підрозділи, з розрахунком загальної кількості штатних одиниць та сумарний оклад. Необхідно передбачити додавання/видалення посад та підрозділів.
- 11) Реалізувати дерево розбору виразу, заданого наведеними нижче синтаксичними правилами:
- <вираз>::=<простий вираз> | <складний вираз>
 - <простий вираз>::=<константа> | <змінна>
 - <константа>::=(\langle число \rangle)
 - <змінна>::=(\langle ім'я \rangle)
 - <складний вираз>::=(\langle вираз \rangle <знак операції><вираз \rangle)
 - <знак операції>::=+|-|*|/
- 12) Реалізувати задачу "Годинник". Повинна бути реалізована можливість використання як годинника зі стрілками, так і цифрового годинника. У класі "Годинники зі стрілками" повинні зберігатися повороти стрілок
- 13) Реалізувати задачу "Комп'ютер". У проекті має бути реалізований "комп'ютер", який виконує основні функції, наприклад, включення,

виключення, запуск ОС, запуск програми, тощо, не розкриваючи клієнту деталей виконання цієї операції

- 14) Реалізувати задачу «Керування домашнім кінотеатром». Домашній кінотеатр може включати наступні компоненти: широкоформатний телевізор, DVD-програвач, ресивер з акустичною системою. Кожен з компонентів має свій інтерфейс управління та ряд налаштувань (наприклад, телевізор – формат відображення, режим яскравості тощо; ресивер – гучність; тощо). Реалізувати загальний інтерфейс керування домашнім кінотеатром, котрий передбачає включення та комплекс налаштувань типових режимів перегляду/прослуховування (наприклад, спорт, кінофільми, класична музика тощо)
- 15) Реалізувати задачу «Піца». В продажу маємо декілька різновидів піци. Однак існують додаткові опції, різні ботики піци, додаткові наповнювачі (наприклад, гриби, помідори, маслини, тощо) . Необхідно при продажу піци динамічно визначати кількість додаткових параметрів та вартість кінцевого продукту.
- 16) Створити набір класів, котрі реалізують наступні перетворення текстових даних с наступним збереженням в файл:
 - а. Переклад по словам з використанням Google Translate API (робочий приклад можна знати, наприклад, тут <https://stackoverflow.com/questions/2246017/using-google-translate-in-c-sharp>)
 - б. Виправлення реєстру символів в реченнях (перша літера – завжди велика, інші - ні).В програмному застосунку повинні бути реалізовані функції запису текстових даних в файл після використання будь-якої комбінації перетворень в будь-якій послідовності та читання даних з файлу зі зворотнім перетворенням.
- 17) Розробити структуру зберігання тексту, який може складатися з абзаців, речень, лістингів, заголовків, слів, знаків пунктуації тощо.
- 18) Розробити структуру організації дискового простору, де каталоги містять інші каталоги та файли. Кожний об'єкт має атрибут розміру.
- 19) Реалізувати завдання «Тварина» і різні його види. За допомогою шаблонів задати способи проживання та харчування.
- 20) Реалізувати завдання «Транспортні засоби». В якості підкласів можуть виступати Автомобіль, Велосипед, Кінь. Підрахувати час і вартість перевезення пасажирів і вантажів кожним транспортним засобом.
- 21) Реалізувати задачу «Пасажироперевізники (транспортні засоби)». Задати правила вибору транспорту в залежності від відстані та наявності шляхів сполучення.

- 22) Реалізувати задачу «Учні навчального закладу». В якості навчального закладу можуть виступати як школи, так і університети. Визначити способи навчання і можливості його продовження.
- 23) Реалізувати задачу «Оркестр». Визначити правила організації (набір музичних інструментів) і управління оркестром.
- 24) Реалізувати завдання «Сад», в якому можуть рости садові дерева (наприклад, яблуні, груші і т.д.). Прийняти рішення про щеплення дерев в залежності від віку і плодоношення. Врахувати, що деякі варіанти схрещування видів неможливі.
- 25) Існує модель системи «Рецепт». Модель дозволяє в незмінному вигляді зберігати призначення лікаря і термін дії рецепту. Реалізувати задачу, що дозволяє продовжувати термін дії вже існуючого рецепту.
- 26) Реалізувати завдання «Перекладач» з англійської мови на українську. Користувач вводить слово англійською мовою і в результаті отримує його переклад українською. Програмне забезпечення повинно мати можливість працювати зі словником раніше збережених слів (зберігати, завантажувати) з файлу. ПЗ повинно мати можливість додавати нові слова.
- 27) Створити набір класів, котрі реалізують наступні перетворення текстових даних с наступним збереженням в файл:
- а. Виправлення регістру символів в реченнях (перша літера – завжди велика, інші - ні).
 - б. Переведення всіх літер у реченні верхній регістр.
- В програмному застосунку повинні бути реалізовані функції запису текстових даних в файл після використання будь-якої комбінації перетворень в будь-якій послідовності та читання даних з файлу зі зворотнім перетворенням.